

# Neural Control for Constrained Human-Robot Interaction with Human Motion Intention Estimation and Impedance Learning

Xinbo Yu

*School of Automation and Electrical Engineering  
University of Science and Technology Beijing  
Beijing, China  
yuxinbo17@qq.com*

Wei He\*

*School of Automation and Electrical Engineering  
University of Science and Technology Beijing  
Beijing, China  
weihe@ieee.org*

Yanan Li

*School of Engineering and Informatics  
University of Sussex  
Brighton, UK  
hit.li.yn@gmail.com*

Chenguang Yang

*College of Automation Science and Engineering  
South China University of Technology  
Guangzhou, China  
cyang@ieee.org*

Changyin Sun

*School of Automation  
Southeast University  
Nanjing, China  
cysun@seu.edu.cn*

**Abstract**—In this paper, an impedance control strategy is proposed for a rigid robot collaborating with human by considering impedance learning and human motion intention estimation. The least square method is used in human impedance identification, and the robot can adjust its impedance parameters according to human impedance model for guaranteeing compliant collaboration. Neural networks (NNs) are employed in human motion intention estimation, so that the robot follows the human actively and human partner costs less control effort. On the other hand, the full-state constraints are considered for operational safety in human-robot interactive processes. Neural control is presented in the control strategy to deal with the dynamic uncertainties and improve the system robustness. Simulation results are carried out to show the effectiveness of the proposed control design.

**Index Terms**—neural networks (NNs), motion intention estimation, adaptive control, impedance learning, full-state constraints, human-robot interaction (HRI).

## I. INTRODUCTION

Driven by the practical needs, robots are usually employed to ease the human workload and reduce the risk of fatigue working. In many complex working procedures, robots can not meet the working requirement without the assistance of human perception. Physical human-robot interaction, combining human's and robot's complementary advantages, plays a key role in many emerging assembly and production processes.

In human-robot interaction, one important problem is how to make the robot collaborate with its human partner efficiently. Impedance control is introduced for compliant interaction because of its robustness and feasibility [1], [2],

[3]. In [4], impedance learning is proposed for the robot to track an unknown desired impedance model, and time-varying impedance parameters are considered to achieve a certain interaction performance. Researchers also develop abundant control methods for impedance learning [5], [6], [7]. If the robots are able to understand the human motion intention, they will be submissive to be operated and human will cost less energy to move the robot to finish tasks. How to estimate the human motion intention attracts substantial attentions from researchers [8], [9].

The operational safety is also important when human interacts with robots. Barrier Lyapunov functions are used to make the robot subject to motion constraints [10], [11], [12]; and input constraints are considered in control design [13], [14]. Neural networks (NNs) are usually employed to handle uncertainties in the models [15], [16]. In [17], an adaptive fuzzy control design and a nonlinear disturbance observer are proposed in the design of a human upper limb exoskeleton. NNs can be not only applied to handle uncertainties in robot models, but also solve the problem about the human motion intention estimation and impedance parameters learning as mentioned above.

The construction of this paper is shown as follows: in Section II, the dynamics of the rigid robot are presented, and meanwhile the control objective is described; in Section III, the human motion intention estimation method is explained in the first place, impedance learning is investigated afterwards, and then the full-state constraints are considered when the robot tracks the virtual trajectory in the inner control loop; in Section IV, the simulations are designed to show the constraint satisfaction and to show the effectiveness of the proposed control design.

This work was supported by the National Natural Science Foundation of China under Grant 61522302, 61761130080, 61520106009, 61533008, the Newton Advanced Fellowship from The Royal Society, UK, under Grant NA160436, the Beijing Natural Science Foundation under Grant 4172041, and the Fundamental Research Funds for the China Central Universities of USTB under Grant FRF-BD-16-005A and FRF-TP-15-005C1.

## II. PROBLEM FORMULATION

### A. System Description

We consider a human-robot interaction scenario where the human collaborates with the robot to perform object transporting. In the human-robot collaborative processes, human will lead the tasks by applying the interaction force on the force sensing handle mounted at the end-effector. The robot's dynamic model in the Cartesian space can be written as [13]

$$M_t(x)\ddot{x} + C_t(x, \dot{x})\dot{x} + G_t(x) = u + f, \quad (1)$$

where  $x, \dot{x}, \ddot{x} \in \mathbb{R}^m$  denote the position, velocity and acceleration vectors in the Cartesian space, respectively.  $M_t(x)$  denotes the inertia matrix,  $C_t(x, \dot{x})$  denotes the Coriolis and centripetal force matrix,  $G_t(x)$  denotes the gravitational force matrix,  $u$  denotes the control force vector, and  $f$  denotes the vector of the interaction force exerted by the human partner.

### B. Control Objective

The control objective is to make the interaction force small in a constrained human-robot interactive process in the presence of unknown impedance model and unknown motion intention of the human partner. In particular, a target impedance model can be considered as

$$f = \Lambda_d(\ddot{x}_d - \ddot{x}_r) + D_d(\dot{x}_d - \dot{x}_r) + K_d(x_d - x_r), \quad (2)$$

where  $\Lambda_d$  denotes the desired inertia matrix,  $D_d$  is the desired damper matrix and  $K_d$  is the desired stiffness matrix.  $x_d$  is the desired trajectory, namely the human motion intention, and  $x_r$  is the virtual desired trajectory. When trajectory tracking control is developed to make  $x$  track  $x_r$  as  $t \rightarrow \infty$ , the impedance model can be rewritten as

$$f = \Lambda_d(\ddot{x}_d - \ddot{x}) + D_d(\dot{x}_d - \dot{x}) + K_d(x_d - x). \quad (3)$$

## III. CONTROL DESIGN

### A. Motion Intention Estimation

The dynamics of the human arm can be described as

$$f = -D_h\dot{x} + K_h(x_d - x), \quad (4)$$

where  $D_h$  and  $K_h$  denote human arm impedance parameter matrices. If  $x_d$  is far away from  $x$ , human partner will cost more effort to move the robot because interaction force  $f$  is significant, and on the contrary, if the robot is aware of the human motion intention  $x_d$ , the human partner will feel easy to operate the robot because of the less interaction force.  $x_d$  can be estimated based on the robot's current position  $x$ , the velocity  $\dot{x}$  and the interaction force  $f$  as below

$$\hat{x}_d = Y(f, x, \dot{x}). \quad (5)$$

Obviously, the function  $Y(\cdot)$  is unknown and may be nonlinear, so a neural network learning method is employed to solve the problem. As an important type of NNs, the radial basis function neural network (RBFNN) is employed to estimate the human motion intention

$$\hat{x}_d = \hat{\Theta}_i^T S(n_i) + \epsilon_i, \quad (6)$$

where  $n_i = [f^T, x^T, \dot{x}^T]$  is the input to RBFNN,  $\Theta_i$  is the actual weight,  $S(\cdot)$  is the radial basis function, and  $\epsilon_i$  is the estimation error. The estimation objective is to make interaction force  $f_i$  as small as possible, where  $f_i$  denotes the interaction force in every direction in the Cartesian space, so  $f = [f_1, f_2, \dots, f_i]$ . The cost function  $E$  based on the steepest descent method can be written as

$$E = \frac{1}{2}f_i^2. \quad (7)$$

We design the adaptation law as

$$\begin{aligned} \dot{\hat{\Theta}}_i &= -\beta_i \frac{\partial E_i}{\partial \hat{\Theta}_i} \\ &= -\beta_i \frac{\partial E_i}{\partial f_i} \frac{\partial f_i}{\partial x_d} \frac{\partial x_d}{\partial \hat{\Theta}_i}, \end{aligned} \quad (8)$$

where  $\beta_i$  is a positive scalar, according to (4), (6) and (7), we obtain

$$\begin{aligned} \frac{\partial E_i}{\partial f_i} &= f_i \\ \frac{\partial f_i}{\partial x_d} &= K_{hi} \\ \frac{\partial x_d}{\partial \hat{\Theta}_i} &= S(n_i). \end{aligned} \quad (9)$$

Then, we have

$$\begin{aligned} \dot{\hat{\Theta}}_i(t) &= -\beta_i f_i K_{hi} S(n_i) \\ &= -\gamma_i f_i S(n_i), \end{aligned} \quad (10)$$

where  $\beta_i$  and  $K_{hi}$  are absorbed by  $\gamma_i$ , so the updating law of  $\hat{\Theta}_i$  is designed as

$$\hat{\Theta}_i(t) = \hat{\Theta}_i(0) - \gamma_i \int_0^t [f_i(\omega) S(n_i)(\omega)] d\omega. \quad (11)$$

Based on the updating law, the estimated motion estimation  $\hat{x}_d$  can be acquired.

### B. Impedance Learning

Note that  $D_h$  and  $K_h$  are unknown impedance parameter matrices. If we have the estimates of  $D_h$ ,  $K_h$  and  $x_d$ , the estimated interaction force  $\hat{f}$  can be written as

$$\hat{f} = -\hat{D}_h\dot{x} + \hat{K}_h z, \quad (12)$$

where  $\hat{D}_h$  and  $\hat{K}_h$  are the estimates of  $D_h$  and  $K_h$ , and  $z = x - \hat{x}_d$ . According to the least square method, we consider the least square cost function  $R$  as

$$R = \sum_{j=1}^n (f_j - \hat{f}_j)^2, \quad (13)$$

where  $j$  denotes the sampling number. The objective of the least square method is to find the solutions of  $D_h$  and  $K_h$  by making  $R$  minimal, so we make partial derivatives of  $R$  with respect to  $D_h$  and  $K_h$  zero, respectively

$$\frac{\partial R}{\partial D_h} = 0, \quad \frac{\partial R}{\partial K_h} = 0. \quad (14)$$

According to the moving average algorithm [18], we obtain

$$\begin{bmatrix} \hat{D}_h(t) \\ \hat{K}_h(t) \end{bmatrix} = \begin{bmatrix} -\sum_{j=t-T+1}^t \dot{x}_j^2 & \sum_{j=t-T+1}^t \dot{x}_j z_j \\ -\sum_{j=t-T+1}^t z_j \dot{x}_j & \sum_{j=t-T+1}^t z_j^2 \end{bmatrix}^{-1} \times \begin{bmatrix} \sum_{j=t-T+1}^t \dot{x}_j f_j \\ \sum_{j=t-T+1}^t z_j f_j \end{bmatrix}, \quad (15)$$

where  $T$  denotes the sampling period. According to the above method, we can obtain the estimation of the human's time-varying impedance parameter matrices  $\hat{D}_h(t)$  and  $\hat{K}_h(t)$ .

When the human is willing to operate the robot by physical human-robot interaction, it would be desirable that the robot reduces the impedance parameters to make it compliant. When the human does not intend to lead the task, the robot will increase the desired impedance parameters to improve the positioning accuracy. Suppose that impedance matrices  $D_d$ ,  $K_d$ ,  $D_h$ ,  $K_h$  are diagonal matrixes. An impedance tuning rule is designed as

$$\begin{aligned} D_h + D_d &= AI_{m \times m} \\ K_h + K_d &= BI_{m \times m}, \end{aligned} \quad (16)$$

where  $A = \text{diag}[a_1, a_2, \dots, a_m]$ ,  $B = \text{diag}[b_1, b_2, \dots, b_m]$ ,  $a_1, a_2, \dots, a_m$  and  $b_1, b_2, \dots, b_m$  are given positive constants. According to the impedance tuning rule, desired impedance parameters can be adjusted to adapt to different situations according to the human impedance variation.

### C. Tracking Control with Full-state Constraints

#### I. Model-based (MB) Control

Let  $x_1 = x$ ,  $x_2 = \dot{x}$ , we have the description of the robot dynamics

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= M(x)^{-1}(u - f(t) - C(x, \dot{x})x_2 - G(x)) \\ y &= x_1. \end{aligned} \quad (17)$$

Define the tracking error as

$$\begin{aligned} e_1 &= x_1 - x_r \\ e_2 &= x_2 - \alpha_1, \end{aligned} \quad (18)$$

where  $\alpha_1$  is a virtual stabilization variable. When  $x$  tracks the virtual desired trajectory  $x_r$ , the robot will track the desired impedance model (3). Provided the initial states  $x_1(0)$  and  $x_2(0)$  are in bounds, all signals are bounded and full-state constraints are not violated, i.e., for  $\forall t > 0$ ,  $|x_{1i}(t)| \leq k_{1i}$ ,  $|x_{2i}(t)| \leq k_{2i}$ .  $k_1$  and  $k_2$  are positive constant vectors which represent the predefined constraints [19]. Choose a barrier Lyapunov function (BLF) candidate  $V_1$  as

$$V_1 = \frac{1}{2} \sum_{i=1}^n \ln \frac{k_{ai}^2}{k_{ai}^2 - e_{1i}^2}, \quad (19)$$

where the tracking error constraints  $k_a = k_1 - \hat{k}_1$ ,  $k_b = k_2 - \hat{k}_2$ ,  $k_{ai} = [k_{a1}, k_{a2}, \dots, k_{an}]$ ,  $k_{bi} = [k_{b1}, k_{b2}, \dots, k_{bn}]$ .  $\hat{k}_1$  and  $\hat{k}_2$  are the vectors composed of the maximum absolute values of

$x_{ri}$  and  $\alpha_{1i}$ . They can be expressed as  $x_{ri} \leq X_{1i}$ ,  $\alpha_{1i} \leq X_{2i}$ . Then differentiating  $V_1$  with respect to time, we have

$$\dot{V}_1 = \sum_{i=1}^n \frac{e_{1i} \dot{e}_{1i}}{k_{ai}^2 - e_{1i}^2}. \quad (20)$$

Differentiating  $e_1$  with respect to time, we have

$$\dot{e}_1 = e_2 + \alpha_1 - \dot{x}_r. \quad (21)$$

We define  $\alpha_1$  in (18) as

$$\alpha_1 = \dot{x}_r - A. \quad (22)$$

The matrix  $A$  denotes

$$A = \begin{bmatrix} (k_{a1}^2 - e_{11}^2)g_{11}e_{11} \\ (k_{a2}^2 - e_{12}^2)g_{12}e_{12} \\ \dots \\ (k_{an}^2 - e_{1n}^2)g_{1n}e_{1n} \end{bmatrix}, \quad (23)$$

where  $g_{1i}$  denotes the gain parameters. Substituting (21), (22) and (23) into (20), we can obtain

$$\dot{V}_1 = -\sum_{i=1}^n g_{1i}e_{1i}^2 + \sum_{i=1}^n \frac{e_{1i}e_{2i}}{k_{ai}^2 - e_{1i}^2}. \quad (24)$$

Then choose a BLF candidate  $V_2$  as

$$V_2 = V_1 + \frac{1}{2}e_2^T M(x)e_2 + \frac{1}{2} \sum_{i=1}^n \ln \frac{k_{bi}^2}{k_{bi}^2 - e_{2i}^2}. \quad (25)$$

Differentiating  $V_2$  with respect to time, we have

$$\begin{aligned} \dot{V}_2 &= -\sum_{i=1}^n g_{1i}e_{1i}^2 + \sum_{i=1}^n \frac{e_{1i}e_{2i}}{k_{ai}^2 - e_{1i}^2} + \sum_{i=1}^n \frac{e_{2i}\dot{e}_{2i}}{k_{bi}^2 - e_{2i}^2} \\ &\quad + e_2^T [u - f(t) - C(x, \dot{x})\alpha_1 - G(x) - M(x)\dot{\alpha}_1]. \end{aligned} \quad (26)$$

Differentiating  $e_2$  with respect to time, we have

$$\begin{aligned} \dot{e}_2 &= \ddot{x}_2 - \dot{\alpha}_1 \\ &= M(x)^{-1}(u - f(t) - C(x, \dot{x})x_2 - G(x)) - \dot{\alpha}_1. \end{aligned} \quad (27)$$

Then we can get the MB control  $u_m$  as

$$\begin{aligned} u_m &= -G_2 e_2 + C(x, \dot{x})\alpha_1 + G(x) + M(x)\dot{\alpha}_1 + f(t) \\ &\quad - (e_2^T)^+ \sum_{i=1}^n \frac{e_{1i}e_{2i}}{k_{ai}^2 - e_{1i}^2} - (e_2^T)^+ \sum_{i=1}^n \frac{e_{2i}(a_i - \dot{\alpha}_{1i})}{k_{bi}^2 - e_{2i}^2}, \end{aligned} \quad (28)$$

where  $G_2$  is the gain matrix,  $(e_2^T)^+$  is the Moore-Penrose inverse of  $e_2^T$ . According to the Moore-Penrose inverse property we can obtain

$$e_2^T (e_2^T)^+ = \begin{cases} 0, & e_2 = [0, 0, \dots, 0]^T \\ 1, & \text{otherwise.} \end{cases} \quad (29)$$

Then, the control input in the joint space is

$$\tau_m = J(q)^T u_m. \quad (30)$$

With the control  $u_m$  in (28), the BLF candidate  $V_2$  satisfies

$$V_2 = -\sum_{i=1}^n k_{1i} e_{1i}^2 - e_2^T G_2 e_2 < 0. \quad (31)$$

Therefore tracking errors  $e_1$  and  $e_2$  remain in the interval  $\forall t > 0, -k_{ai} \leq e_{1i} \leq k_{ai}, -k_{bi} \leq e_{2i} \leq k_{bi}$ .

## II. Adaptive Neural Network Control with Full-State Feedback

Due to the uncertainties in robot dynamics, an adaptive NN control is designed in this part. The adaptive law is designed as

$$\dot{\hat{W}}_i = -\Gamma_i [S_i(Z_i) e_{2i} + \sigma_i \hat{W}_i], i = 1, 2, \dots, n \quad (32)$$

where  $\hat{W}_i$  is the weight estimate of NN,  $\Gamma_i = \Gamma_i^T$  is the positive gain matrix and  $\sigma_i$  is a small positive constant which is used to improve the system robustness.  $Z_i = [x_1^T, x_2^T, \alpha^T, \dot{\alpha}^T]$  are the inputs of NN.  $\hat{W}^T S(Z)$  is used to estimate  $W^{*T} S(Z)$ , as below

$$\begin{aligned} W^{*T} S(Z) &= \hat{W}^T S(Z) - \epsilon(z) \\ &= C(x_1, x_2) \alpha_1 + G(x_1) + \dot{\alpha}_1 M(x_1) - \epsilon(z), \end{aligned} \quad (33)$$

where  $W_i^*$  is the actual weight of NN, the estimation error  $\epsilon(z)$  is in bounds over the compact set  $\Omega$ ,  $\forall Z \in \Omega, |\epsilon(z)| < \bar{\epsilon}$ , with  $\bar{\epsilon}$  as a positive constant. We propose the NN control input  $u$  as

$$\begin{aligned} u &= -G_3 e_2 + \hat{W}^T S(Z) + f - (e_2^T)^+ \sum_{i=1}^n \frac{e_{1i} e_{2i}}{k_{ai}^2 - e_{1i}^2} \\ &\quad - (e_2^T)^+ \sum_{i=1}^n \frac{e_{2i} (a_i - \dot{\alpha}_{1i})}{k_{bi}^2 - e_{2i}^2} - (e_2^T)^+ \sum_{i=1}^n \frac{g_{1i} e_{1i}^2}{k_{ai}^2 - e_{1i}^2} \\ &\quad - (e_2^T)^+ \sum_{i=1}^n \frac{g_{2i} e_{2i}^2}{k_{bi}^2 - e_{2i}^2}, \end{aligned} \quad (34)$$

where  $g_{2i}$  denotes the gain parameter,  $G_3$  is the gain matrix which is positive definite. The NN control in the joint space is

$$\tau = J(q)^T u. \quad (35)$$

Consider another Lyapunov function  $V_3$  as

$$V_3 = V_2 + \frac{1}{2} \sum_{i=1}^n \tilde{W}_i^T \Gamma_i^{-1} \tilde{W}_i, \quad (36)$$

where  $\tilde{W}_i$  denotes the weight error. Differentiating  $V_3$  yields

$$\begin{aligned} \dot{V}_3 &= -\sum_{i=1}^n g_{1i} e_{1i}^2 + \sum_{i=1}^n \frac{e_{1i} e_{2i}}{k_{ai}^2 - e_{1i}^2} + \sum_{i=1}^n \frac{e_{2i} \dot{e}_{2i}}{k_{bi}^2 - e_{2i}^2} \\ &\quad + e_2^T [u - f(t) - C(x, \dot{x}) \alpha_1 - G(x) - M(x) \dot{\alpha}_1]. \end{aligned} \quad (37)$$

We obtain

$$\begin{aligned} \dot{V}_3 &\leq -\sum_{i=1}^n g_{1i} e_{1i}^2 - e_2^T G_2 e_2 - \sum_{i=1}^n \frac{g_{1i} e_{1i}^2}{k_{ai}^2 - e_{1i}^2} - e_2^T \epsilon(z) \\ &\quad - \sum_{i=1}^n \frac{g_{2i} e_{2i}^2}{k_{bi}^2 - e_{2i}^2} + e_2^T \hat{W}^T S(Z) - e_2^T W^{*T} S(Z) \\ &\quad + \sum_{i=1}^n \tilde{W}_i^T \Gamma_i^{-1} \{-\Gamma_i [S_i(Z) e_{2i} + \sigma_i \hat{W}_i]\} \\ &\leq -e_2^T (G_2 - I) e_2 - \sum_{i=1}^n \frac{g_{1i} e_{1i}^2}{k_{ai}^2 - e_{1i}^2} - \sum_{i=1}^n \frac{g_{2i} e_{2i}^2}{k_{bi}^2 - e_{2i}^2} \\ &\quad + \frac{1}{2} \|\epsilon(z)\|^2 + \frac{\sigma_i}{2} (\|W_i^*\|^2 - \|\tilde{W}_i\|^2) \\ &\leq -\rho V_3 + C, \end{aligned} \quad (38)$$

where

$$\begin{aligned} \rho &= \min(\min(2g_{1i}), \min(2g_{2i}), \frac{2\lambda_{\min}(G_2 - I)}{\lambda_{\max}(M)}, \\ &\quad \min(\frac{\sigma_i}{\Gamma_i^{-1}})) \\ C &= \frac{1}{2} \|\bar{\epsilon}\|^2 + \frac{\sigma_i}{2} \|W_i^*\|^2. \end{aligned} \quad (39)$$

where  $\lambda$  denotes the eigenvalue of a matrix.

**Theorem 1:** For each compact set  $\Omega_0$ , the initial conditions  $x_{1i}(0)$  and  $x_{2i}(0)$  are in bounds, the control ( $x_{1i}$  tracks  $x_{ri}$ ,  $x_{2i}$  tracks  $\alpha_{1i}$ ) with  $|x_{1i}(0)| \leq k_{1i}, |x_{2i}(0)| \leq k_{2i}$  guarantees that the closed-loop system is semi-globally uniformly bounded (SGUB) and full-state constraints are not transgressed,  $\forall t > 0, |x_{1i}(t)| \leq k_{1i}, |x_{2i}(t)| \leq k_{2i}$ . The close-loop error signals  $e_1, e_2$  remain in the compact sets,  $\Omega_{e_1}, \Omega_{e_2}$ , and the weight error  $\tilde{W}$  remains in the compact set  $\Omega_{\tilde{W}}$ , i.e.,

$$\begin{aligned} \Omega_{e_1} &= \{e_1 \in R^n \mid \|e_1\| \leq \sqrt{k_{ai}^2 (1 - e^{-D})}, i = 1, 2, 3, \dots, n\} \\ \Omega_{e_2} &= \{e_2 \in R^n \mid \|e_2\| \leq \sqrt{\frac{D}{\lambda_{\min}(M)}}\} \cap \\ &\quad \{e_2 \in R^n \mid \|e_2\| \leq \sqrt{k_{bi}^2 (1 - e^{-D})}, i = 1, 2, 3, \dots, n\} \\ \Omega_{\tilde{W}} &= \{\tilde{W} \in R^{l \times n} \mid \|\tilde{W}\| \leq \sqrt{\frac{D}{\lambda_{\min}(\Gamma^{-1})}}\}, \end{aligned} \quad (40)$$

## IV. SIMULATION

We consider a two link revolute robot in the task space and an interaction force is applied on the end effector by the human partner.  $m_r$  and  $l_r$  denote the mass and length of link  $r$ , respectively.  $l_{cr}$  denotes the distance from joint  $r-1$  to the mass center of link  $r$ , and  $I_r$  denotes link  $r$ . The simulation parameter values are chosen as:  $m_1=2.0\text{kg}$ ,  $m_2=0.85\text{kg}$ ,  $l_1=1.40\text{m}$ ,  $l_2=1.24\text{m}$ ,  $l_{c1}=0.70\text{m}$ ,  $l_{c2}=0.62\text{m}$ ,  $I_1=0.980\text{kgm}^2$ ,  $I_2=0.953\text{kgm}^2$ .

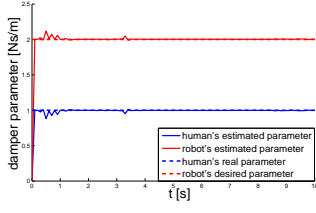


Fig. 1. Case A: fixed human impedance

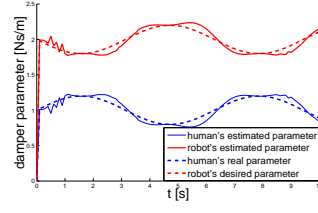


Fig. 2. Case A: Time-varying human impedance

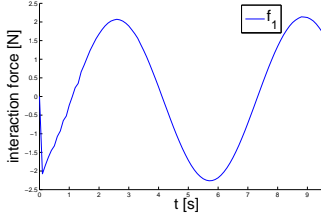


Fig. 3. Case B: Interaction force

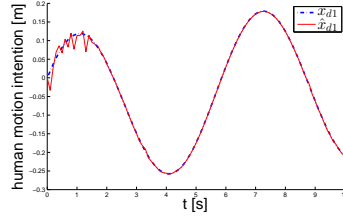


Fig. 4. Case B: Human motion intention estimation

### A. Impedance learning simulation

Two cases about the human arm impedance learning are considered. The human arm impedance parameters are fixed but they are unknown for the robot in the first case, which are  $D_h = \text{diag}[1, 1]$  and  $K_h = \text{diag}[2, 2]$ . The parameters in the impedance tuning rule  $A = \text{diag}[3, 3]$  and  $B = \text{diag}[3, 3]$ . In the second case, we consider that the human arm impedance parameters are time-varying, i.e.,  $D_h = \text{diag}[1 + 0.2\sin(t), 1 + 0.2\sin(t)]$ , and the other parameters are the same as in the first case. As indicated in Fig. 1 and Fig. 2, the fixed or time-varying human arm's damper parameters can be estimated and the robot's desired damper parameters can be learned by the proposed method. Similarly, we can obtain the stiffness parameters by the impedance learning method.

### B. Motion estimation simulation

As for the human motion intention estimation, we use the NN to estimate the human motion intention. The RBF centers are chosen in the region of  $[-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1]$ , the NN node is chosen as  $2^6$ , the initial value of the NN weights  $\Theta_i$  is set as 0. The impedance parameter matrices of human arm are chosen as  $D_h = \text{diag}[0.1, 0.1]$ ,  $K_h = \text{diag}[2, 2]$ . In Fig. 3, the human motion intention estimation is involved and the initial human motion intention  $x_{d1}(0)$  is set as 10.35m, and the interaction forces are below 2N in human robot interactive processes. In Fig. 4, the  $x_{d1}$  and  $\hat{x}_{d1}$  denotes the real desired trajectory and the human motion intention estimation in the x-axis in the Cartesian space. From this figure, we can see that the estimated error between  $\hat{x}_{d1}$  and  $x_{d1}$  is converging to zero.

### C. Full-state constraints simulation

We evaluate two control methods to show the effectiveness of the proposed full-state constraints method, which are NN

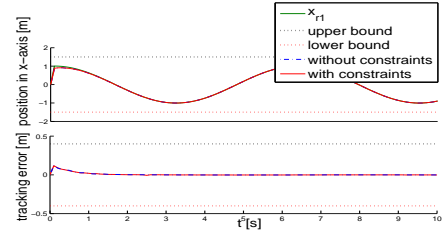


Fig. 5. Case C: position tracking in x-axis

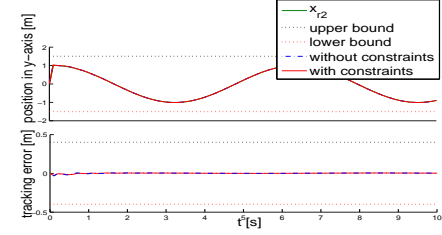


Fig. 6. Case C: position tracking in y-axis

control with full-state constraints and NN control without full-state constraints.

For the control method with full-state constraints, we set the tracking error constraints as  $k_{a1} = 0.4\text{m}$ ,  $k_{a2} = 0.4\text{m}$ ,  $k_{b1} = 0.4\text{m/s}$ ,  $k_{b2} = 0.4\text{m/s}$  and the full-state constraint vectors  $k_1 = [1.5\text{m}; 1.5\text{m}]$ ,  $k_2 = [1.5\text{m/s}; 1.5\text{m/s}]$ ; the gain parameters are chosen as  $g_{11} = 2$ ,  $g_{12} = 2$ ,  $g_{21} = 10$ ,  $g_{22} = 10$ ,  $G_3 = \text{diag}[10, 10]$ , and the simulation time  $t$  is set as 10s. Fig. 5 and Fig. 6 show the control performance with and without full-state constraints about the position tracking, respectively. Fig. 7 and Fig. 8 show the velocity tracking performance with and without constraints, respectively. According to these figures, for the NN control with full-state constraints, we can conclude that if the initial states are in bounds, all the signals are not transgressed to the constraints, and the tracking errors are converging to a small field around zero. For the control without constraints, we can find that although the state variables can track the desired trajectory, the tracking errors without constraints are larger than those under the control with full-state constraints.

### D. Synthetic simulation

The robot moves from the initial positions  $x_1(0)=[0\text{m}, 0\text{m}]$ , and the initial velocities  $x_2(0)=[0\text{m/s}, 0\text{m/s}]$ . In this part, we

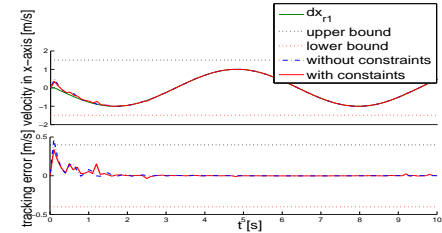


Fig. 7. Case C: velocity tracking in x-axis

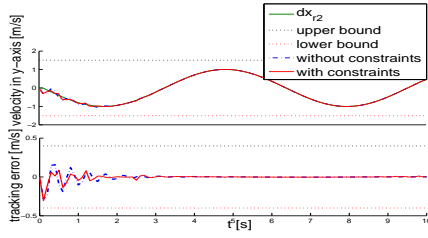


Fig. 8. Case C: velocity tracking in y-axis

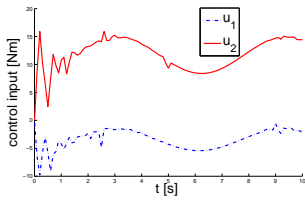


Fig. 9. Case D: control input

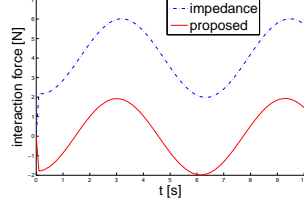


Fig. 10. Case D: interaction force

consider both impedance learning and human motion intention estimation, subject to the full-state constraints,  $\forall t > 0$ ,  $-k_{ai} \leq e_{1i} \leq k_{ai}$ ,  $-k_{bi} \leq e_{2i} \leq k_{bi}$ , where  $k_{a1} = 0.4\text{m}$ ,  $k_{a2} = 0.4\text{m}$ ,  $k_{b1} = 0.4\text{m/s}$ ,  $k_{b2} = 0.4\text{m/s}$ . The robot's virtual reference trajectory  $x_r$  is given as

$$x_r = \begin{bmatrix} (0.1\sin(t) + \cos(t))\text{m} \\ (0.1\sin(t) + \cos(t))\text{m} \end{bmatrix}. \quad (41)$$

For the proposed NN control, the gain parameters are chosen as  $g_{11} = 2$ ,  $g_{12} = 2$ ,  $g_{21} = 5$ ,  $g_{22} = 5$ ,  $K_3 = \text{diag}[10, 10]$ , the RBF centers are chosen in the region of  $[-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1]$ , the NN node is chosen as  $2^8$ , the initial value of the NN weight is set as 0.  $\Gamma_1$  and  $\Gamma_2$  are selected as 100, and  $\sigma_i = 0.002$ . The corresponding control input is shown in Fig. 9. In Fig. 10, if the motion intention estimation is not involved and the desired trajectory is fixed in the traditional impedance control design, the interaction force will be larger than the force under the proposed control design considering the human motion intention estimation.

## V. CONCLUSION

In this paper, an impedance control strategy is proposed for a rigid robot collaborating with human by considering impedance learning and human motion intention estimation. The least square method is used in human impedance identification, and the robot can adjust its impedance parameters according to human impedance model for guaranteeing the compliant collaboration. NNs are employed in human motion intention estimation, so that the robot follows the human actively and human partner costs less control effort. On the other hand, the full-state constraints are considered for operational safety in human-robot interactive processes. Neural control is presented in the control strategy to deal with the dynamic uncertainties and improve the system robustness. Eventually,

the simulation results are carried out to show the effectiveness of the proposed control design.

## REFERENCES

- [1] Y. Li, S. S. Ge, and C. Yang, "Learning impedance control for physical robot-environment interaction," *International Journal of Control*, vol. 85, no. 2, pp. 182–193, 2012.
- [2] P. H. Chang, K. Park, S. H. Kang, H. I. Krebs, and N. Hogan, "Stochastic estimation of human arm impedance using robots with nonlinear frictions: An experimental validation," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 2, pp. 775–786, 2013.
- [3] Y. Li and S. S. Ge, "Human-robot collaboration based on motion intention estimation," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 3, pp. 1007–1014, 2014.
- [4] Y. Li and S. S. Ge, "Impedance learning for robots interacting with unknown environments," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1422–1432, 2014.
- [5] D. Wang and C. Cheah, "An iterative learning- control scheme for impedance control of robotic manipulators," *The International Journal of Robotics Research*, vol. 17, no. 10, pp. 1091–1104, 1998.
- [6] B. Kim, J. Park, S. Park, and S. Kang, "Impedance learning for robotic contact tasks using natural actor-critic algorithm," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 2, pp. 433–443, 2010.
- [7] T. Tsuji, K. Ito, and P. G. Morasso, "Neural network learning of robot arm impedance in operational space," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 26, no. 2, pp. 290–298, 1996.
- [8] M. S. Erden and T. Tomiyama, "Human-intent detection and physically interactive control of a robot without force sensors," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 370–382, 2010.
- [9] K. Wakita, J. Huang, P. Di, K. Sekiyama, and T. Fukuda, "Human-walking-intention-based motion control of an omnidirectional-type cane robot," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 1, pp. 285–296, 2013.
- [10] W. He, A. O. David, Z. Yin, and C. Sun, "Neural network control of a robotic manipulator with input deadzone and output constraint," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 6, pp. 759–770, 2016.
- [11] W. He, Y. Dong, and C. Sun, "Adaptive neural impedance control of a robotic manipulator with input saturation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 3, pp. 334–344, 2016.
- [12] M. Chen, "Robust tracking control for self-balancing mobile robots using disturbance observer," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 3, pp. 458–465, 2017.
- [13] Z. Li, S. S. Ge, M. Adams, and W. S. Wijesoma, "Robust adaptive control of uncertain force/motion constrained nonholonomic mobile manipulators," *Automatica*, vol. 44, no. 3, pp. 776–784, 2008.
- [14] Z. Fu, W. Xie, S. Rakheja, and J. Na, "Observer-based adaptive optimal control for unknown singularly perturbed nonlinear systems with input constraints," *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 1, pp. 48–57, 2017.
- [15] C. Yang, Y. Jiang, Z. Li, W. He, and C. Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 3, pp. 1162–1171, 2017.
- [16] C. Yang, Z. Li, R. Cui, and B. Xu, "Neural network-based motion control of an underactuated wheeled inverted pendulum model," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 11, pp. 2004–2016, 2014.
- [17] Z. Li, C.-Y. Su, L. Wang, Z. Chen, and T. Chai, "Nonlinear disturbance observer-based control design for a robotic exoskeleton incorporating fuzzy approximation," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 9, pp. 5763–5775, 2015.
- [18] G. Xu, A. Song, L. Pan, X. Gao, Z. Liang, J. Li, and B. Xu, "Clinical experimental research on adaptive robot-aided therapy control methods for upper-limb rehabilitation," *Robotica*, vol. 32, no. 7, pp. 1081–1100, 2014.
- [19] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Transactions on Cybernetics*, vol. 46, no. 3, pp. 620–629, 2016.